

This is G o o g l e's cache of <http://developer.apple.com/documentation/mac/Toolbox/Toolbox-450.html>.  
 G o o g l e's cache is the snapshot that we took of the page as we crawled the web.  
 The page may have changed since that time. Click here for the [current page](#) without highlighting.  
 To link to or bookmark this page, use the following url: <http://www.google.com/search?q=cache:E2xsfx53lfEJ:developer.apple.com/documentation/mac/Toolbox/Toolbox-450.html++%22separate+file+++%22++application+%22+&hl=en&ie=UTF-8>

*Google is not affiliated with the authors of this page nor responsible for its content.*

These search terms have been highlighted: **separate file application**



Developer Connection

Search

[Advanced Search](#)



[Log In](#) | [Not a Member?](#)

[Contact ADC](#)

[Previous](#)

[Book Contents](#)

[Book Index](#)

[Next](#)

[Inside Macintosh: Macintosh Toolbox Essentials /](#)  
[Chapter 7 - Finder Interface / Using the Finder Interface](#)

## Creating File Reference Resources

File reference ('FREF') resources perform two main functions. First, they associate icons you define with file types used by your **application**. Second, they allow users to drag document icons to your **application** icon in order to open them from your **application**.

Create a file reference resource for your **application** file itself and create **separate file** reference resources for each file type that your **application** can open. [Listing 7-3](#) shows, in Rez input format, the file reference resources for the SurfWriter **application** file, text documents, stationery pads, and editions and for TeachText read-only documents.

Each file reference resource specifies the following items:

- a file type
- the local ID of an icon list resource as assigned in the bundle resource
- an empty string

The file type can be defined for files created by your **application** only, for files created by other applications that your **application** supports, or for files of the existing general types, such as 'TEXT' and 'PICT'.

As described in the next section, "[Creating a Bundle Resource](#)," the local ID maps the file type to an icon list resource that is assigned the same local ID in the bundle resource. If you wanted two file types to share the same icon, for example, you could create two

**separate file** reference resources that share the same local ID, which the bundle resource would map to the same icon list resource. (Creating two file types that share the same icon is not recommended, however, because a shared icon would make it very difficult for the user to distinguish between the different file types while using the Finder.)

### Listing 7-3 Rez input for file reference resources

```
resource 'FREF' (208, purgeable) { /*SurfWriter application*/
    'APPL', /*type 'APPL'*/
    0, /*maps to icon list resource w/ local ID 0 in bundle resource*/
    "" /*leave empty string for name: not implemented*/
};
resource 'FREF' (209, purgeable) { /*SurfWriter document*/
    'TEXT', /*type 'TEXT'*/
    1, /*maps to icon list resource w/ local ID 1 in bundle resource*/
    ""
};
resource 'FREF' (210, purgeable) { /*SurfWriter stationery pad*/
    'sEXT', /*type 'sEXT'*/
    2, /*maps to icon list resource w/ local ID 2 in bundle resource*/
    ""
};
resource 'FREF' (211, purgeable) { /*SurfWriter edition*/
    'edtt', /*type 'edtt'*/
    3, /*maps to icon list resource w/ local ID 3 in bundle resource*/
    ""
};
resource 'FREF' (212, purgeable) { /*TeachText read-only files*/
    'ttro', 4, "" /*These documents have TeachText as their */
                /* creator. Finder uses TeachText's icon list resource */
                /* for these documents. Included here so users */
                /* can drag these docs to SurfWriter's app icon*/
};
```

If you provide your own icon for the stationery pads that users create from your **application's** documents, create a file reference resource for your stationery pads. Assign this file reference resource a file type in the following manner: use the file type of the document upon which the stationery pad is based, but replace the first letter of the original document's file type with a lowercase *s*. As with other file reference resources, you map this to an icon list resource in the bundle resource. (This convention necessitates that you make the names of your documents' file types unique in their last three letters.)

For example, in [Listing 7-3](#), the 'sEXT' file type assigned within the file reference resource is used for stationery pads created from documents of the 'TEXT' file type. In this case, when the `isStationery` bit (described in ["Using Finder Information in the Catalog File" beginning on page 7-30](#)) is set on a document of file type 'TEXT', the Finder looks in the SurfWriter **application's** bundle ('BNDL') resource to determine what icon is mapped to documents of type 'sEXT'. The Finder then displays the document using the stationery pad icon shown in Plate 4 at the front of this book.

When the user drags a document icon to your **application** icon, the Finder checks a list that it maintains of your file reference resources. If the document's file type appears in this list, the Finder launches your **application** with a request to open that document.

If your **application** supports file types for which it doesn't provide icons, you can still define file reference resources for them, and then users can launch your **application** by dragging these document icons to your **application** icon. For example, the file reference resource with resource ID 212 in [Listing 7-3 on page 7-18](#) is created so that the Finder launches the SurfWriter **application** when users drag TeachText read-only documents to the SurfWriter **application** icon. Since these documents have TeachText as their creator, the Finder displays the icon that the TeachText **application** defines for them in its own bundle resource.

By supporting the Open Documents event, you can also specify disks, folders, and a pair of wildcard file types in your file reference resources so that users can launch your **application** by dragging their icons to your **application** icon. As explained in *Inside Macintosh: Interapplication Communication*, the Open Documents event is one of the four required Apple events. After the Finder uses the Process Manager to launch an **application** that supports high-level events, the Finder sends your **application** an Open Documents event, which includes a list of alias records for objects that the **application** should open.

Because alias records can specify volumes and directories as well as files, an Open Documents event gives you the opportunity to handle cases in which users drag disk or folder icons to your **application**. (Alias records are described in ["Using Aliases" beginning on page 7-37](#).) Create a file reference resource and specify 'disk' as the file type to allow users to drag hard disk and floppy disk icons to your **application** icon. Create a file reference resource and specify 'fold' as the file type to allow users to drag folder icons to your **application** icon.

You can create a file reference resource that specifies '\*\*\*\*\*' as the file type to allow users to drag all file types--including applications, system extensions, documents, and so on, but not including disks or folders--to your **application** icon. If you create three file reference resources that specify 'disk', 'fold', and '\*\*\*\*\*' as their file types and if your **application** supports the Open Documents event, you effectively allow users to launch your **application** by dragging any icon to your **application** icon. It is up to your **application** to open disks, folders, or all possible file types in a manner appropriate to the needs of the user.

---

[Previous](#)[Book Contents](#)[Book Index](#)[Next](#)

© Apple Computer, Inc.  
11 JUL 1996

Get information on [Apple](#) products.  
Visit the Apple Store [online](#) or at [retail](#) locations.  
1-800-MY-APPLE

Copyright © 2003 Apple Computer, Inc.  
[All rights reserved.](#) | [Terms of use](#) | [Privacy Notice](#)